

Máster Avanzado de Programación en Python para Hacking, BigData y Machine Learning

Buenas Prácticas de Programación en Python

LECCIÓN 03

Programación dirigida por la documentación

ÍNDICE

Introducción

Objetivos

Documentación: método tradicional

Documentación con Sphinx

Conclusiones

INTRODUCCIÓN

En esta lección aprenderemos a llevar a cabo una programación dirigida por una correcta documentación.

Documentar el código de forma adecuada no solo facilita la tarea de terceros a la hora de comprender nuestras implementaciones, sino que también nos sirve de utilidad a nosotros mismos para depurar y analizar nuestro código.

OBJETIVOS

Al finalizar esta lección serás capaz de:

- 1 Conocer las herramientas disponibles en Python para documentar nuestro código.
- 2 Aprender buenas prácticas para documentar nuestro código.
- 3 Aprender con un caso práctico a documentar un programa escrito en Python.

DOCUMENTACIÓN: MÉTODO TRADICIONAL

Documentar un código de programación consiste en explicar detalladamente en qué consiste nuestro programa, qué funciones la componen y cómo un usuario puede utilizar nuestro código.

En términos generales podemos decir que la documentación es un conjunto de instrucciones que ayudan a un usuario a utilizar cada una de las funciones que hemos implementado en un determinado programa.

El hecho de realizar una documentación de calidad es soportado por varios motivos, siendo el más importante el compromiso del creador de Python, Guido van Rossum, con el desarrollo de documentación sobre el propio lenguaje Python, así como sus librerías.

DOCUMENTACIÓN: MÉTODO TRADICIONAL

El compromiso de toda una comunidad de desarrolladores de Python permite desde la creación de informes de error a simples reclamaciones cuando la documentación realizada por otros usuarios podría ser más completa o fácil de utilizar.

Este tipo de formas de comunicación y retroalimentación ha demostrado una gran eficacia a la hora de encontrar códigos documentados correctamente.

La documentación realizada en Python se basa en Latex y exige un conjunto de macros especialmente escritos para documentar Python correctamente.

DOCUMENTACIÓN CON SPHINX

Sphinx es una herramienta desarrollada por Georg Barndt con el objetivo de facilitar la tarea de documentar proyectos software.

Esta herramienta ha sido desarrollada especialmente para crear documentación inteligente y atractiva.

Inicialmente fue creada para crear documentación de proyectos Python aunque actualmente puede ser utilizado con otros lenguajes de programación.

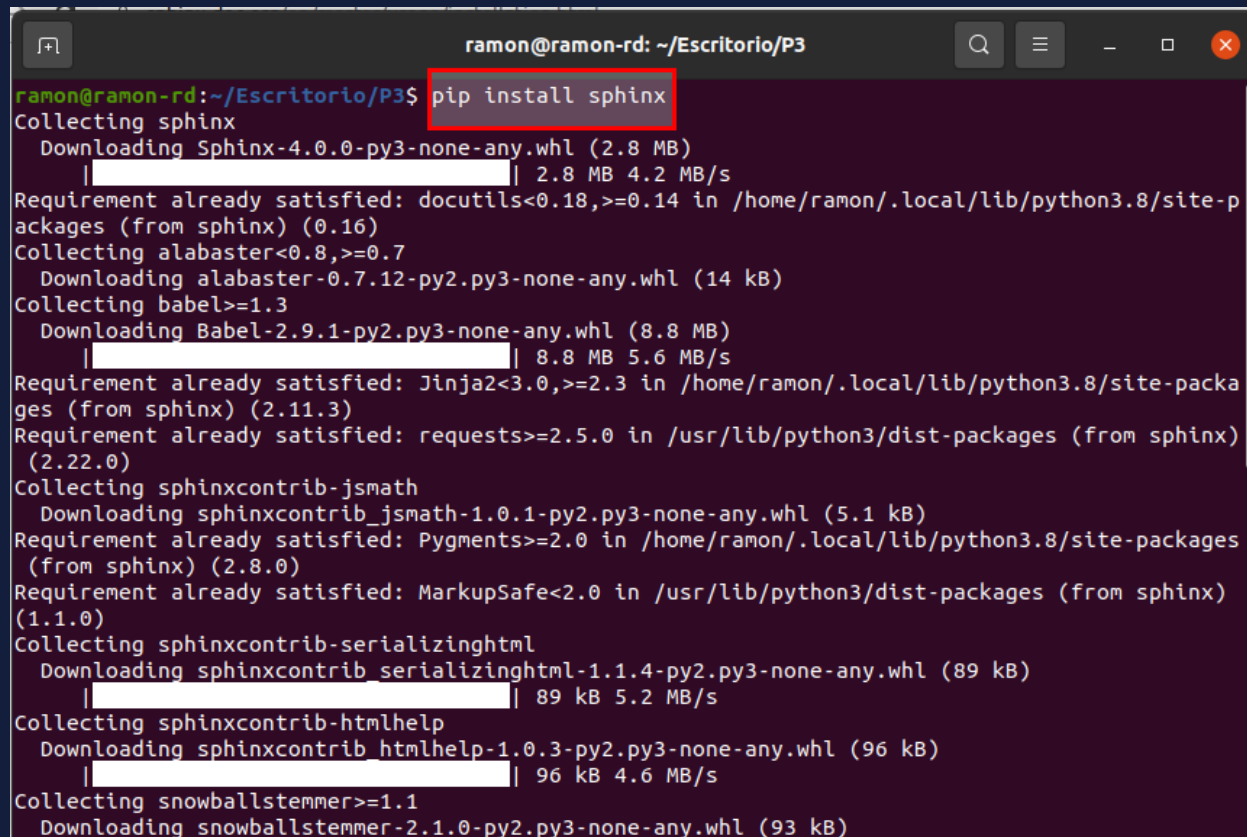
DOCUMENTACIÓN CON SPHINX

Sphinx cuenta con las siguientes características:

- Gran variedad de formatos de salida: HTML, LaTeX, Epub, etc.
- Referencias cruzadas.
- Estructura jerárquica.
- Índices automáticos.
- Resaltado automático de código.
- Extensiones.

DOCUMENTACIÓN CON SPHINX

Instalación:



```
ramon@ramon-rd: ~/Escritorio/P3
ramon@ramon-rd:~/Escritorio/P3$ pip install sphinx
Collecting sphinx
  Downloading Sphinx-4.0.0-py3-none-any.whl (2.8 MB)
    | 2.8 MB 4.2 MB/s
Requirement already satisfied: docutils<0.18,>=0.14 in /home/ramon/.local/lib/python3.8/site-packages (from sphinx) (0.16)
Collecting alabaster<0.8,>=0.7
  Downloading alabaster-0.7.12-py2.py3-none-any.whl (14 kB)
Collecting babel>=1.3
  Downloading Babel-2.9.1-py2.py3-none-any.whl (8.8 MB)
    | 8.8 MB 5.6 MB/s
Requirement already satisfied: Jinja2<3.0,>=2.3 in /home/ramon/.local/lib/python3.8/site-packages (from sphinx) (2.11.3)
Requirement already satisfied: requests>=2.5.0 in /usr/lib/python3/dist-packages (from sphinx) (2.22.0)
Collecting sphinxcontrib-jsmath
  Downloading sphinxcontrib_jsmath-1.0.1-py2.py3-none-any.whl (5.1 kB)
Requirement already satisfied: Pygments>=2.0 in /home/ramon/.local/lib/python3.8/site-packages (from sphinx) (2.8.0)
Requirement already satisfied: MarkupSafe<2.0 in /usr/lib/python3/dist-packages (from sphinx) (1.1.0)
Collecting sphinxcontrib-serializinghtml
  Downloading sphinxcontrib_serializinghtml-1.1.4-py2.py3-none-any.whl (89 kB)
    | 89 kB 5.2 MB/s
Collecting sphinxcontrib-htmlhelp
  Downloading sphinxcontrib_htmlhelp-1.0.3-py2.py3-none-any.whl (96 kB)
    | 96 kB 4.6 MB/s
Collecting snowballstemmer>=1.1
  Downloading snowballstemmer-2.1.0-py2.py3-none-any.whl (93 kB)
```

DOCUMENTACIÓN CON SPHINX

Caso práctico...





CONCLUSIONES

1

Documentar un proyecto en Python es crucial para ayudar a la comunidad de programadores a entender nuestro código.

2

La tarea de documentar un código correctamente no es un ejercicio trivial. Requiere constancia y disciplina para llevar a cabo una documentación de calidad.

3

Debido a su complejidad, podemos encontrar diversas herramientas que nos ayudan con esta tarea.

MUCHAS GRACIAS POR SU ATENCIÓN



rrueda@grupomainjobs.com



Ramón Rueda Delgado
<https://www.linkedin.com/in/ramon-rueda/>



twitter.com/eiposgrados



facebook.com/eiposgrados



instagram.com/eiposgrados